

FIG. 1

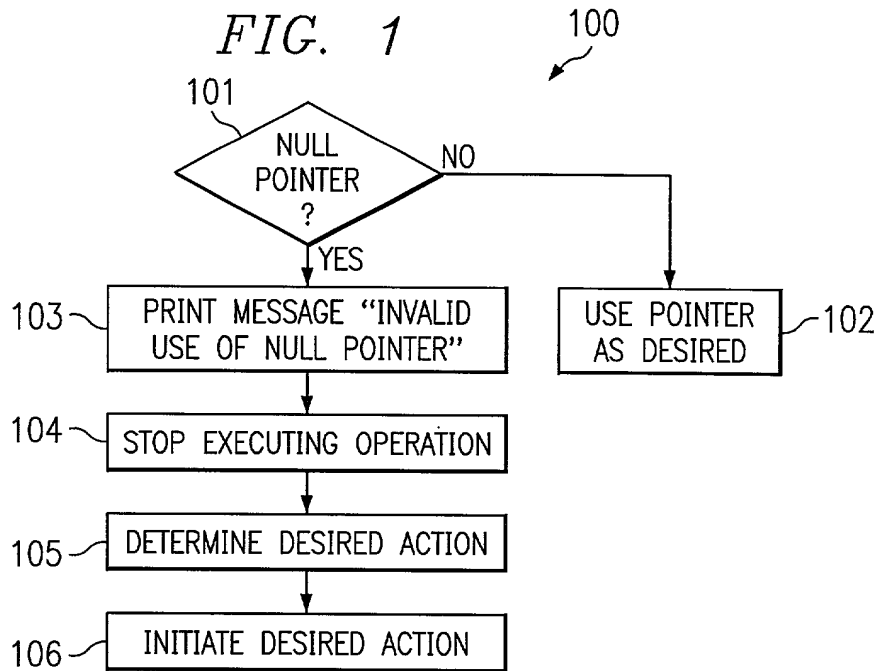


FIG. 2

```

201 ~ #include <iostream.h>
203 ~ template <class C>
204 ~ class SafePtr {
205 ~     public:
206 ~         SafePtr (C *p) : ptr(p) {}
207 ~         operator C *() const { return ptr; }
208 ~         C *operator-> () const {
209 ~             if (!ptr) {
210 ~                 cerr << "\nAttempted indirection of "
211 ~                     << "NULL pointer.\n" ;
212 ~                 exit(1); // or throw an exception
213 ~             }
214 ~             return ptr;
215 ~         }
216 ~     private:
217 ~         C *const ptr;
218 ~ };
  
```

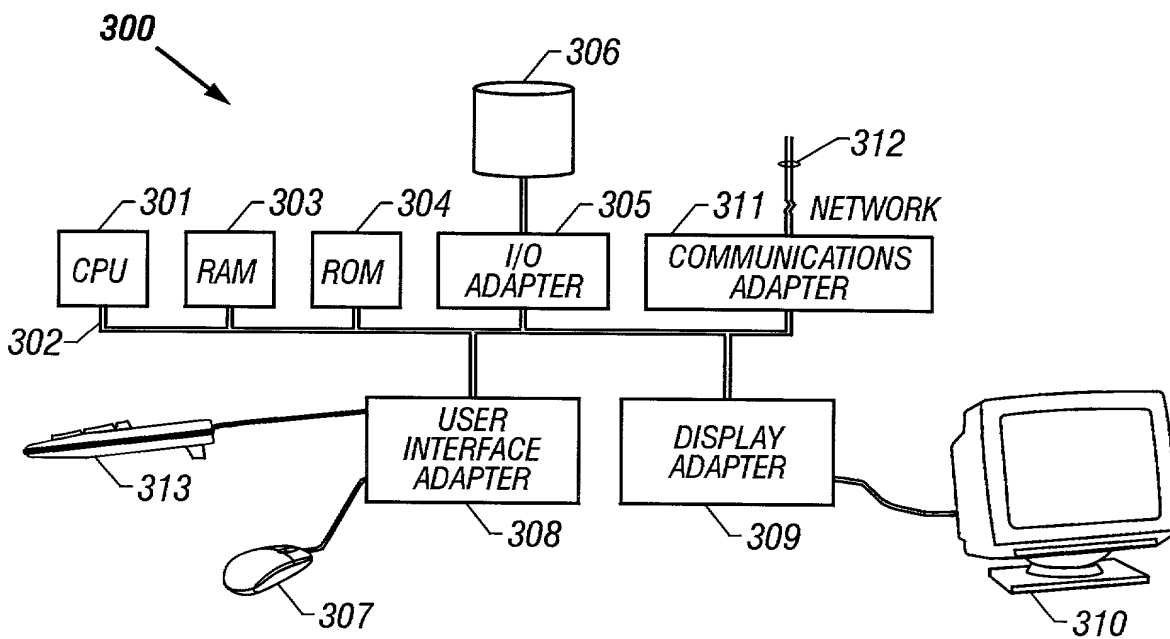


FIG. 3